

Inventory Planner Manual

Getting Started

Welcome to the **Inventory Planner** documentation.

Whether you are a new or an advanced user, you can find some useful information here.

Go ahead, dive in!

Firstly, please, find our extension in [My Downloadable Products](#) section of our store. Learn [how to install extension](#), and proceed with [Configuration](#), which will help you to set up your **Inventory Planner** service.

How to install the extension

How to install the extension using composer

1. Backup your store's database and web directory.
2. Login to the SSH console on your server and navigate to the root directory of the Magento 2 store.
3. Copy the installation instructions from the page [My Downloadable Products](#) to the SSH console and press ENTER.
4. Run command `php -f bin/magento module:enable Mirasvit_Core Mirasvit_InventoryPlanner` to enable the extension.
5. Run command `php -f bin/magento setup:upgrade` to install the extension.
6. Run command `php -f bin/magento cache:clean` to clean the cache.
7. Deploy static view files

```
rm -rf pub/static/*; rm -rf var/view_preprocessed/*; php -f bin/magento setup:static-content:deploy
```

How to install the extension manually

1. Backup your store's database and web directory.
2. Download archive from [My Downloadable Products](#).
3. Unzip the extension locally.
4. Copy unzipped folder to the root directory of the Magento store.
5. Run command `php -f bin/magento module:enable Mirasvit_Core Mirasvit_InventoryPlanner` to enable the extension.
6. Run command `php -f bin/magento setup:upgrade` to install the extension.
7. Run command `php -f bin/magento cache:clean` to clean the cache.

8. Deploy static view files

```
rm -rf pub/static/*; rm -rf var/view_preprocessed/*; php -f  
bin/magento setup:static-content:deploy
```

?

Inventory Planner

The extension Inventory Planner starts working right after installation without any manual reconfiguration required. Still it provides the option for setting up the module for your particular use case.

The Inventory Planner updates its predictions right after any change of the inventory data.

Configuring settings

The Inventory Planner extension will make the forecasts on its own directly after the installation. For that, it has default values which can be edited for your particular case.

To find settings of the Inventory Planner extension, navigate to **Stores > Settings > Configuration > Mirasvit Extensions > Inventory Planner**

Configuration

- GENERAL
- CATALOG
- SECURITY
- CUSTOMERS
- SALES
- YOTPO
- DOTDIGITAL
- MIRASVIT EXTENSION 1**
- Inventory Planner
- Developer 1
- SERVICES

Forecast Settings

- Sales history period (days) [global]

Stock Settings

- Lead Time (days) [global]
- Stock Cover Time (days) [global]
- Overstock Time (days) [global]
- Overstock qty threshold [global]

You can change the **Forecast Settings** and **Stock Settings**.

Forecast Settings has one parameter:

- Sales history period (in days). Used for calculation Sales Velocity.

Stock Settings have following parameters:

- Lead Time (days). The average amount of time it takes to order a product from the supplier and receive it into inventory.
- Stock Cover Time (days). The average number of days that actual inventory quantity should cover.
- Overstock Time (days). A product is considered overstock when actual stock quantity is greater than overstock time (used with Sales Velocity)
- Overstock qty threshold. Ignore overstock if product quantity is less than this value.

Using the Inventory Planner

The Inventory Planner has three pages: Inventory, Replenishment, Overstock. Basically these serve as one product grid which is reconfigured for convenient depiction of information on a separate page for each task.

On any of the three pages, you can add the following columns regarding the each product in the inventory to the grid:

- actual stock quantity
- is in stock (yes/no)
- sale velocity (items/day)
- sells out in (days)
- replenishment quantity
- replenishment in (days)
- overstock quantity
- last sell date
- sales for 7, 30, 90 days, total sales
- lead time
- days of stock (on the module settings page, it is called Stock Cover Time)
- quantity at each warehouse

You can filter products in the grid by the values in these columns.

Inventory

To look at the current inventory status and inventory forecast in the Magento admin panel sidebar, navigate to **Stores > Inventory Planner > Inventory**. Renew the index by clicking the **Update** button.

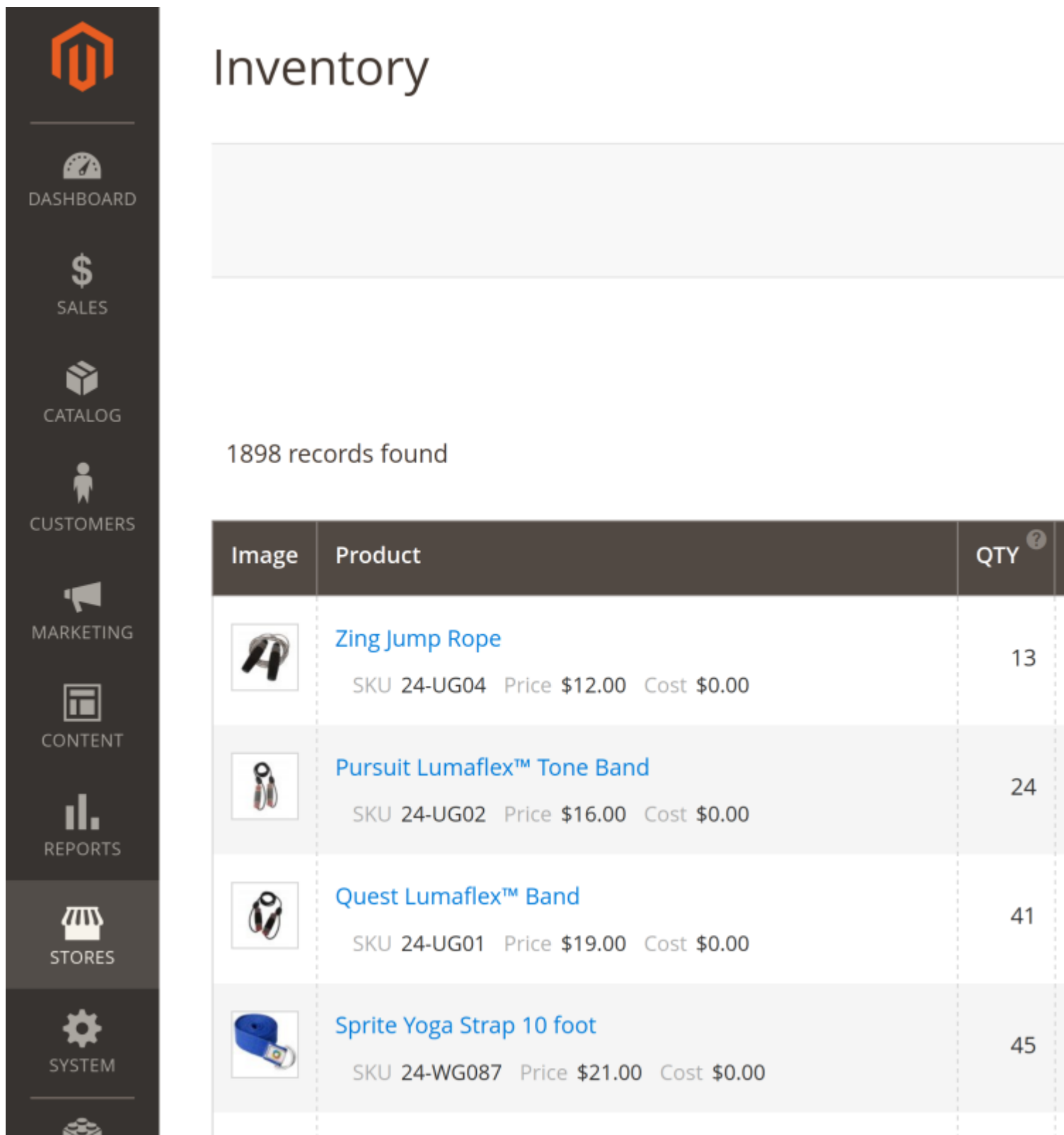






Image	Product	QTY [?]
	Zing Jump Rope SKU 24-UG04 Price \$12.00 Cost \$0.00	13
	Pursuit Lumaflex™ Tone Band SKU 24-UG02 Price \$16.00 Cost \$0.00	24
	Quest Lumaflex™ Band SKU 24-UG01 Price \$19.00 Cost \$0.00	41
	Sprite Yoga Strap 10 foot SKU 24-WG087 Price \$21.00 Cost \$0.00	45






The inventory page is capable of displaying the sales velocity and predicting when the available stock will be depleted. For the latter, it uses such parameters as Lead Time and Days of Stock (on the module settings page, it is called Stock Cover Time).

The **Sells Out In** value is calculated as the actual stock quantity (Qty) divided by the Sales Velocity. The result decimal fraction is rounded down to the nearest integer.

Replenishment

The Replenishment page allows you to quickly see the what items are needed to be ordered from the supplier. The extension automatically calculates when and how many items you should order.

Navigate to **Stores > Inventory Planner > Replenishment** using the Magento admin panel sidebar. Click the **Update** button to renew the grid index.

Image	Product	QTY [?]
	Zing Jump Rope SKU 24-UG04 Price \$12.00 Cost \$0.00	13
	Juno Jacket-XL-Green SKU WJ06-XL-Green Price \$77.00 Cost \$0.00	8
	Olivia 1/4 Zip Light Jacket-XL-Purple SKU WJ12-XL-Purple Price \$77.00 Cost \$0.00	23
	Pursuit Lumaflex™ Tone Band SKU 24-UG02 Price \$16.00 Cost \$0.00	24
	Josie Yoga Jacket-XL-Blue SKU WJ02-XL-Blue Price \$56.25 Cost \$0.00	32

The replenishment proposition is also calculated using the **Stock Cover Time** value which is defined in the module settings.

The **Replenishment In** is determined as the **Lead Time** subtracted from the **Sells Out In** time. The result decimal fraction is rounded up to the nearest integer.

The replenishment quantity is calculated as:

Replenishment Quantity = (Replenishment In * Sales Velocity) + (Sales Velocity

When the **Replenishment In** is greater than the **Sells Out In**, the module shows the **Replenishment In** and **Replenishment Qty** as zero.

Overstock

Navigate to **Stores > Inventory Planner > Overstock** using the Magento admin panel sidebar to see the overstock information. Click the **Update** button to renew the grid index.



Overstock



DASHBOARD



SALES



CATALOG



CUSTOMERS



MARKETING



CONTENT








REPORTS



STORES

1885 records found

Image	Product	QTY
	Joust Duffle Bag SKU 24-MB01 Price \$34.00 Cost \$0.00	1
	Strive Shoulder Pack SKU 24-MB04 Price \$32.00 Cost \$0.00	1
	Crown Summit Backpack SKU 24-MB03 Price \$38.00 Cost \$0.00	1
	Wayfarer Messenger Bag SKU 24-MB05 Price \$45.00 Cost \$0.00	1
	Rival Field Messenger	1

The module detects goods that sell too slowly and automatically estimates an overstock quantity based on the Overstock Time and Overstock qty threshold parameters in its settings.

How to upgrade extension

To upgrade the extension, follow these steps:

1. Backup your store's database and web directory.

2. Login to the SSH console of your server and navigate to the root directory of the Magento 2 store.
3. Run command `composer require mirasvit/module-inventory-planner:* --update-with-dependencies` to update current extension with all dependencies.

Note

In some cases, the command above is not applicable, or it's not possible to update just the current module, or you need to upgrade all Mirasvit modules in a bundle. In this case, the command above will have no effect.

Run instead `composer update mirasvit/*` command. It will update all Mirasvit modules installed in your store.

4. Run command `php -f bin/magento setup:upgrade` to install updates.
5. Run command `php -f bin/magento cache:clean` to clean the cache.
6. Deploy static view files

```
rm -rf pub/static/*; rm -rf var/view_preprocessed/*; php -f bin/magento setup:static-content:deploy
```

Disabling the Extension

Temporarily Disable

To temporarily disable the extension please follow these steps:

1. Login to the SSH console on your server and navigate to the root directory of the Magento 2 store.
2. Run the command `php -f bin/magento module:disable Mirasvit_InventoryPlanner` to disable the extension.
3. Log in to the Magento backend and refresh the store cache (if enabled).

Extension Removal

To uninstall the extension, please follow these steps:

1. Login to the SSH console on your server and navigate to the root directory of the Magento 2 store.
2. Run command `php -f bin/magento module:disable Mirasvit_InventoryPlanner` to disable the extension.
3. Run command `composer remove mirasvit/module-inventory-planner` to remove the extension.
4. Log in to the Magento backend and refresh the store cache (if enabled).

1.2.6

(2026-04-21)

Improvements

- Added i18n support for 15 core locales (ar_SA, cs_CZ, de_DE, es_ES, fr_FR, it_IT, ja_JP, nl_NL, pl_PL, pt_BR, pt_PT, sv_SE, tr_TR, uk_UA, zh_Hans_CN)
-

1.2.5

(2026-04-17)

Features

- **Configurable sales period windows** — the "Sold Qty" time windows shown in Inventory Planner grids are now configurable. Go to Stores ? Configuration ? Inventory Planner ? Forecast Settings ? Sales periods and enter any comma-separated list of day counts (e.g. 7 , 30 , 90 , 180). Default is 7, 30, and 90 days. Useful for tracking slower-moving inventory with longer windows.
 - **Sold Price column in order history** — the order history grid on each product's inventory detail page now shows the actual per-unit price paid (including tax) in the order's currency. This lets buyers see what customers actually paid per unit, rather than relying on current catalog prices.
-

1.2.4

(2026-04-02)

Improvements

- **Faster re-indexing for large catalogs** — the inventory indexer now uses bulk database operations throughout: products are loaded and saved in batches of 500, stock levels and reservations are fetched in a single query per batch (previously one query per SKU), and sales data for all time windows (7-day, 30-day, 90-day, all-time) is retrieved in one query instead of five. Stores with thousands of products will see significantly shorter re-index times.
-

1.2.3

(2025-08-04)

Fixed

- Issue related to missing product attribute values in inventory grids
-

1.2.2

(2025-07-25)

Fixed

- Fixed the issue with data generation from performance toolkit profiles in EE
-

1.2.1

(2025-07-10)

Features

- Added the ability to import/export suppliers and product groups

Improvements

- Added product group column to inventory planner grids
-

1.2.0

(2025-06-17)

Improvements

- WCAG 2.2 AA compliance
-

1.1.18

(2025-04-16)

Fixed

- Issue related to a violation of strict type checking rules
-

1.1.17

(2025-04-15)

Fixed

- Compatibility with PHP 8.4
-

1.1.16

(2025-04-03)

Features

- Added inventory stock changes history
-

1.1.15

(2024-06-25)

Improvements

- Added a Product ID column to grids
-

1.1.14

(2024-04-19)

Improvements

- Added inventory indexer
-

1.1.13

(2023-05-02)

Improvements

- Ability to hide extra columns to improve performance
 - Added additional checks to avoid creating extra rows in the mst_inventory_planner_supplier_product table
-

1.1.12

(2023-03-27)

Fixed

- Number of sales
-

1.1.11

(2022-12-30)

Fixed

- Issue with ACI for adding products to a group or supplier.
-

1.1.10

(2022-12-29)

Fixed

- Acl title
-

1.1.9

(2022-12-16)

Fixed

- Removed grouped products
-

1.1.8

(2022-11-24)

Features

- Product groups
-

1.1.7

(2022-11-15)

Improvements

- Option to enable/disable the display of disabled products
-

1.1.6

(2022-08-25)

Fixed

- Compatibility with Magento 2.4.4
-

1.1.5

(2022-08-16)

Fixed

- Only enabled products are processed
-

1.1.4

(2022-08-01)

Features

- Ability to select order statuses for calculation
-

1.1.3

(2022-07-14)

Fixed

- Filter by Status
-

1.1.2

(2022-06-27)

Improvements

- Export visible columns
-

1.1.1

(2022-06-20)

Improvements

- remove db_schema_whitelist.json
-

1.1.0

(2022-05-25)

Improvements

- Migrate to declarative schema
-

1.0.15

(2021-12-30)

Features

- Ability to assign one product to several suppliers
-

1.0.14

(2021-11-01)

Improvements

- Updated "Sales Velocity" calculation formula. Now the metric related to forecast time or product creation time
-

1.0.13

(2021-10-25)

Improvements

- Added SKU column
-

1.0.12

(2021-10-21)

Fixed

- Compatibility with Magento 2.3.0-2.3.2
-

1.0.11

(2021-10-20)

Improvements

- Added export
-

1.0.10

(2021-07-27)

Improvements

- Added ability to choose allowed sources for calculations
-

1.0.9

(2021-06-30)

Improvements

- Added cronjob to reindex data every day
-

1.0.8

(2021-05-13)

Fixed

- Compatibility issue with Magento Enterprise
-

1.0.7

(2021-04-26)

Improvements

- Added Saleable Qty columns
-

1.0.6

(2021-04-22)

Improvements

- Added Name to the filter
-

1.0.5

(2021-04-19)

Fixed

- QTY to Salable QTY
-

1.0.4

(2021-03-01)

Improvements

- Multi-stock
-

1.0.3

(2021-02-09)

Fixed

- Issue with joining product attributes
-

1.0.1

(2021-02-09)

Improvements

- Ability to display columns in the grid and filter by product attributes
-

1.0.0

(2021-01-12)

Improvements

- Initial Release